

Exercice 1 : Chaînes de caractères

- 1.a]** Écrire un programme qui affiche la table des caractères ASCII.
- 1.b]** Écrire une fonction qui renverse les lettres d'un mot stocké dans une chaîne de caractères.
- 1.c]** Écrire une fonction qui teste si un mot est un palindrome par exemple "laval".
- 1.d]** Écrire une fonction qui décode une date de naissance codée selon le format "AAAA-MM-JJ", en retournant :
- Jour du mois : ??
 - Mois : ??
 - Année : ????
 - Puis, l'âge de la personne

On suppose que l'entrée est bien formatée. On ne demande pas l'âge exact en fonction de la date courante et de sa date de naissance, mais une indication sur l'année dans laquelle se trouve la personne en fonction de l'année 2009. On pourra écrire "La personne se trouve dans sa 20ième année".

Exercice 2 : Fichiers

- 2.a]** Écrire une fonction qui ouvre un fichier en lecture, et écrit le même contenu mais en majuscule dans un deuxième fichier ouvert en écriture.
- 2.b]** Écrire un programme qui compte le nombre de lignes, de mots, et de caractères d'un fichier.

Exercice 3 : Age exact pour le décodage de la date de naissance

- 3.a]** Calculer exactement l'âge en utilisant la librairie `time.h` pour connaître la date actuelle en manipulant la structure `struct tm` et le type `time_t`. La fonction `time` initialise une variable de type `time_t` passée par adresse avec le nombre de secondes écoulées depuis le 1 janvier 1970. La fonction `localtime` prend cette variable et initialise la structure `struct tm` avec les informations qui vont bien sur l'année, le mois, le jour, etc. Cette fonction `localtime` retourne l'adresse de la structure initialisée. Il suffit ensuite de lire cette structure. Attention, il faut rajouter une quantité sur l'année car l'année est donnée avec un décalage de 1900.

Exercice 4 : Chiffrement de César

Le chiffrement de César consiste à remplacer chaque lettre d'une chaîne de caractères par une autre chaîne de même longueur où chaque caractère est remplacé par un autre caractère situé 3 positions plus loin dans l'alphabet. Par exemple, le caractère **a** est remplacé par **d**, **b** par **e**, **c** par **f**, **d** par **g**, ..., **w** par **z**, **x** par **a**, **y** par **b** et **z** par **c**. On suppose que le message à chiffrer ne contient pas d'espace ou d'apostrophe et qu'il n'est constitué que des lettres de l'alphabet.

4.a] Écrire la fonction de chiffrement qui prend en entrée une chaîne de caractères représentant le message à chiffrer et retourne une autre chaîne de même longueur contenant le message chiffré.

Supposons que l'on renforce ce système de chiffrement en choisissant un décalage aléatoire d compris entre 0 et 25 et en décalant toutes les lettres de d . Par exemple, le système de César a un décalage de 3 et un décalage de 0 correspond à l'identité.

4.b] Proposer une fonction qui renvoie 26 chaînes de caractères qui correspondent aux 26 décalages possibles.

Pour attaquer ce système, c'est-à-dire retrouver le message d'origine sans connaître le décalage, l'idée est d'observer que la lettre **e** est la lettre la plus courante en français. On s'attend donc à voir cette lettre le plus souvent dans le message d'origine si le texte est suffisamment long et que le texte correspond à un texte français normal. Ainsi la lettre qui apparaît le plus souvent dans le message chiffré doit avec forte probabilité correspondre au chiffrement de la lettre **e**. On peut donc en déduire le décalage et retrouver le message d'origine.

4.c] Écrire une fonction qui prend une chaîne de caractères et retourne la lettre qui apparaît le plus souvent.

4.d] Proposer une fonction qui utilise cette idée pour retourner une chaîne correspondant au texte d'origine avec forte probabilité.